Dokumentacja zadania 3.9 (system decyzyjny bota komputerowego)

1. Autor: Mateusz Kiljan
2. Data ukończenia: 06.06.2018r
3. Opis programu: Celem projektu było napisanie systemu decyzyjnego bota komputerowego. W tym celu w programie zaimplementowana została maszyna stanów, która odpowiada za logikę bota (wykonywanie akcji, zmianę stanów). Stworzona została klasa State, po której dziedziczą stany tj. Attack, Defend etc. Do każdego stanu zagregowana została klasa ActionList, która odpowiada za wykonywane akcje np. wyświetlenie na ekranie ciosu, odjęcie HP przeciwnikowi etc. ( akcje te nie zostały zdefiniowane, ponieważ zależą one od konkretnego typu gry). Maszyna posiada klasę Transition, która agreguje obiekty Klas Condition oraz klas pochodnych. Do klasy Transition, zagregowana została publicznie klasa Action, ponieważ nie w każdej maszynie stanów Transition musi wyzwalać akcję (zostaje to do sprecyzowania przez osobę piszącą konkretną maszynę stanów). Zostały napisane trzy testy jednostkowe, których działanie wyświetlane jest w oknie konsoli.
4. Testy sprawdzają:

* Test01 – funkcjonalność klasy Condition oraz klas pochodnych
* Test02 – funkcjonalność klasy Transition
* Test03 – funkcjonalność klasy StateMachine

1. Link do repozytorium na github’ie : https://github.com/matekkil1/bot\_system\_3.9\_project\_mateusz\_kiljan
2. Program można modyfikować poprzez np.:
   * Dodanie nowych stanów
   * Dodawanie dodatkowych atrybutów do każdego ze stanów
   * Dodanie nowych warunków
   * Zaimplementowanie bardziej szczegółowej klasy Action
   * Stworzenie pochodnej klasy po Klasie StateMachine, która będzie sprecyzowana dla konkretnego rodzaju bota komputerowego
   * Implementacja maszyny stanów do gry komputerowej - rozbudowa klas (World, Bot , BotBrain)
3. Uwagi:
   * Klasy: World, Bot , BotBrain są klasami pustymi i mają na celu jedynie pokazanie ewentualnej dalszej rozbudowy programu ( pisanie gry nie było wymagane w treści zadania) dlatego też nie zostały one zawarte w diagramie UML.